AWS Well-Architected 프레임워크: 보안 핵심 요소 브리핑

요약

본 문서는 AWS Well-Architected 프레임워크의 보안 원칙(Security Pillar)에 대한 심층 분석을 제공합니다. 이 프레임워크는 클라우드에서 안정적이고, 안전하며, 효율적이고, 비용 효율적이며, 지속 가능한 워크로드를 설계하고 운영하기 위한 AWS의 현재 모범 사례를 제시합니다. 보안 원칙은 비즈니스 및 규제 요구 사항을 충족하면서 데이터, 시스템 및 자산을 보호하는 데 중점을 둡니다.

핵심 요점:

1. **보안 설계 원칙:** 클라우드 보안을 강화하기 위한 7가지 핵심 원칙을 제시합니다. 여기에는 강력한 자격 증명 기반 구현(최소 권한 원칙), 추적 가능성 유지, 모든 계층에 보안 적용(심층 방어), 보안 모범 사례 자동화, 전송 중 및 저장 데이터 보호, 데이터로부터 사람 분리, 보안 이벤트 대비가 포함됩니다.
2. **책임 공유 모델:** 보안은 AWS와 고객 간의 공유된 책임입니다. AWS는 클라우드 자체의 보안("Security *of* the Cloud")을 책임지며, 하드웨어, 소프트웨어, 네트워킹 및 시설을 보호합니다. 고객은 클라우드 내의 보안("Security *in* the Cloud")을 책임지며, 선택한 서비스에 따라 게스트 운영 체제, 애플리케이션 소프트웨어 및 방화벽 구성을 관리합니다.
3. **보안의 7가지 영역:** 보안 원칙은 7가지 핵심 영역으로 구성됩니다: 보안 기반, 자격 증명 및 액세스 관리, 탐지, 인프라 보호, 데이터 보호, 사고 대응, 애플리케이션 보안. 각 영역은 워크로드의 전체 수명 주기에 걸쳐 보안을 구축하고 유지하기 위한 구체적인 모범 사례를 제공합니다.
4. **자동화의 중요성:** 보안 프로세스, 테스트 및 검증의 자동화는 보안 운영을 확장하고, 일관성을 유지하며, 인적 오류를 줄이는 데 필수적입니다. 인프라를 코드로(IaC) 정의하고 CI/CD 파이프라인을 통해 배포함으로써 변경 사항을 안정적으로 적용하고 구성을 표준화할 수 있습니다.
5. **사전 대비 및 대응:** 성숙한 예방 및 탐지 제어 기능을 갖추더라도, 조직은 보안 사고에 대응하고 잠재적 영향을 완화하기 위한 메커니즘을 구현해야 합니다. 사고 대응 계획, 플레이북 개발, 포렌식 기능 준비, 정기적인 시뮬레이션 실행은 효과적인 복구를 위해 매우 중요합니다.

서론

AWS Well-Architected 프레임워크는 AWS에서 워크로드를 구축하는 동안 내리는 결정의 장단점을 이해하는 데 도움을 줍니다. 이 프레임워크를 사용하면 안정적이고, 안전하며, 효율적이고, 비용 효율적이며, 지속 가능한 워크로드를 설계하고 운영하기 위한 최신 아키텍처 모범 사례를 배울 수 있습니다. 프레임워크는 운영 우수성, 보안, 안정성, 성능 효율성, 비용 최적화, 지속 가능성의 여섯 가지 원칙을 기반으로 합니다.

이 문서는 보안 원칙에 초점을 맞춰, 현재 AWS 권장 사항에 따라 비즈니스 및 규제 요구 사항을 충족하는 데 도움을 줍니다. 최고 기술 책임자(CTO), 최고 정보 보안 책임자(CISO), 아키텍트, 개발자 및 운영 팀 구성원과 같은 기술 역할을 맡은 사람들을 대상으로 합니다. 이 문서를 통해 데이터와 시스템을 보호하고, 액세스를 제어하며, 보안 이벤트에 자동으로 대응하는 아키텍처를 구축하는 방법을 이해할 수 있습니다.

보안 기반

보안 원칙은 클라우드 기술을 활용하여 데이터, 시스템 및 자산을 보호하고 보안 태세를 개선하는 방법을 설명합니다.

설계 원칙

클라우드 워크로드 보안을 강화하는 데 도움이 되는 7가지 핵심 원칙은 다음과 같습니다.

- **강력한 자격 증명 기반 구현:** 최소 권한 원칙을 구현하고 AWS 리소스와의 각 상호 작용에 대해 적절한 권한 부여로 직무 분리를 시행합니다. 자격 증명 관리를 중앙 집중화하고 장기 정적 자격 증명에 대한 의존을 없애는 것을 목표로 합니다.
- **추적 가능성 유지:** 환경에 대한 작업 및 변경 사항을 실시간으로 모니터링, 경고 및 감사합니다. 로그 및 메트릭 수집을 시스템과 통합하여 자동으로 조사하고 조치를 취합니다.
- **모든 계층에 보안 적용:** 여러 보안 제어를 통한 심층 방어 접근 방식을 적용합니다. 네트워크 엣지, VPC, 로드 밸런싱, 모든 인스턴스 및 컴퓨팅 서비스, 운영 체제, 애플리케이션 및 코드 등 모든 계층에 적용합니다.
- **보안 모범 사례 자동화:** 자동화된 소프트웨어 기반 보안 메커니즘은 보다 빠르고 비용 효율적으로 안전하게 확장할 수 있는 능력을 향상시킵니다. 버전 제어 템플릿에서 코드로 정의되고 관리되는 제어 구현을 포함하여 안전한 아키텍처를 만듭니다.
- **전송 중 및 저장 데이터 보호:** 데이터를 민감도 수준으로 분류하고 암호화, 토큰화 및 액세스 제어와 같은 메커니즘을 적절하게 사용합니다.

- **데이터로부터 사람 분리:** 데이터에 대한 직접적인 액세스나 수동 처리의 필요성을 줄이거나 없애는 메커니즘과 도구를 사용합니다. 이는 민감한 데이터를 처리할 때 잘못된 처리나 수정 및 인적 오류의 위험을 줄입니다.
- **보안 이벤트 대비:** 조직의 요구 사항에 맞는 사고 관리 및 조사 정책과 프로세스를 마련하여 사고에 대비합니다. 사고 대응 시뮬레이션을 실행하고 자동화된 도구를 사용하여 탐지, 조사 및 복구 속도를 높입니다.

책임 공유 모델

보안 및 규정 준수는 AWS와 고객 간의 공유된 책임입니다. 이 모델은 AWS가 호스트 운영 체제 및 가상화 계층부터 서비스가 운영되는 시설의 물리적 보안에 이르기까지 구성 요소를 운영, 관리 및 제어하므로 고객의 운영 부담을 덜어줄 수 있습니다. 고객은 게스트 운영 체제(업데이트 및 보안 패치 포함) 및 기타 관련 애플리케이션 소프트웨어와 AWS에서 제공하는 보안 그룹 방화벽의 구성을 책임집니다.

책임 주체	책임 범위
AWS (클라우드 자체의 보안)	하드웨어/AWS 글로벌 인프라(리전, 가용 영역, 엣지 로케이션), 소프트웨어(컴퓨팅, 스토리지, 데이터베이스, 네트워킹)를 보호합니다.
고객 (클라우드 내의 보안)	고객 데이터, 플랫폼, 애플리케이션, 자격 증명 및 액세스 관리, 운영 체제, 네트워크 및 방화벽 구성, 클라이언트 측 데이터 암호화 및 데이터 무결성 인증, 서버 측 암호화, 네트워킹 트래픽 보호를 책임집니다.

IT 제어 또한 이 모델을 따릅니다.

- **상속된 제어:** 고객이 AWS로부터 완전히 상속하는 제어 (예: 물리적 및 환경적 제어).
- **공유된 제어:** 인프라 계층과 고객 계층 모두에 적용되는 제어. AWS는 인프라 요구 사항을 제공하고 고객은 자체 제어 구현을 제공해야 합니다 (예: 패치 관리, 구성 관리, 인식 및 교육).
- **고객 특정 제어:** 고객이 AWS 서비스 내에 배포하는 애플리케이션을 기반으로 전적으로 고객의 책임인 제어 (예: 서비스 및 통신 보호 또는 영역 보안).

AWS 계정 관리 및 분리

워크로드를 별도의 계정으로 구성하고, 조직의 보고 구조를 반영하기보다는 기능, 규정 준수 요구 사항 또는 공통 제어 집합을 기반으로 계정을 그룹화하는 것이 좋습니다. AWS에서 계정은 강력한 경계선 역할을 하므로, 프로덕션 워크로드를 개발 및 테스트 워크로드에서 분리하는 데 강력히 권장됩니다.

- **중앙 집중식 계정 관리:** AWS Organizations를 사용하여 AWS 계정 생성 및 관리를 자동화합니다.
- **중앙 집중식 제어 설정:** 서비스 제어 정책(SCP)을 사용하여 조직, 조직 단위(OU) 또는 계정 수준에서 권한 가드레일을 적용하여 허용되는 서비스, 리전 및 서비스 작업을 제어합니다.
- **중앙 집중식 서비스 및 리소스 구성:** AWS CloudTrail을 사용하여 조직 전체의 모든 작업을 중앙에서 로깅하고, AWS Config를 사용하여 규정 준수 여부를 감사합니다.

자격 증명 및 액세스 관리(IAM)

AWS 서비스를 사용하려면 사용자와 애플리케이션에 AWS 계정의 리소스에 대한 액세스 권한을 부여해야 합니다. 견고한 자격 증명 관리 및 권한을 통해 올바른 사람이 올바른 조건 하에서 올바른 리소스에 액세스할 수 있도록 보장해야 합니다.

자격 증명 관리 모범 사례

- **강력한 로그인 메커니즘 사용 (SEC02-BP01):** 다단계 인증(MFA)을 요구하고 강력한 암호 정책을 시행하여 의도하지 않은 자격 증명 액세스 위험을 줄입니다.
- **임시 자격 증명 사용 (SEC02-BP02):** 장기 자격 증명 대신 임시 자격 증명을 사용하여 자격 증명이 노출, 공유 또는 도난당할 위험을 줄입니다.
- **비밀을 안전하게 저장 및 사용 (SEC02-BP03):** AWS Secrets Manager와 같은 전용 서비스를 사용하여 API 키, 암호, OAuth 토큰과 같은 비밀을 저장, 관리 및 교체합니다.
- **중앙 집중식 자격 증명 공급자 사용 (SEC02-BP04):** 인력(직원, 계약자)의 경우 중앙 자격 증명 공급자를 통해 ID를 관리하여 여러 애플리케이션과 시스템 전반의 액세스 관리를 용이하게 합니다.
- **자격 증명 정기적 감사 및 교체 (SEC02-BP05):** 장기 자격 증명을 정기적으로 감사하고 교체하여 자격 증명이 리소스에 액세스하는 데 사용될 수 있는 기간을 제한합니다.
- **사용자 그룹 및 속성 사용 (SEC02-BP06):** 사용자 그룹 및 속성에 따라 권한을 정의하여 정책의 수와 복잡성을 줄이고 최소 권한 원칙을 더 쉽게 달성합니다.

권한 관리 모범 사례

- **최소 권한 액세스 부여 (SEC03-BP02):** 사용자가 특정 리소스에서 특정 조건 하에 특정 작업을 수행하는 데 필요한 액세스 권한만 부여합니다.

- **비상 액세스 프로세스 수립 (SEC03-BP03):** 중앙 집중식 자격 증명 공급자에 문제가 발생할 경우를 대비하여 워크로드에 비상 액세스를 허용하는 프로세스를 만듭니다.
- **지속적으로 권한 축소 (SEC03-BP04):** 불필요한 권한을 제거하고 검토 프로세스를 수립하여 최소 권한을 달성합니다. 사용되지 않는 ID와 권한을 지속적으로 모니터링하고 제거합니다.
- **조직의 권한 가드레일 정의 (SEC03-BP05):** 서비스 제어 정책(SCP)과 같은 권한 가드레일을 사용하여 주체에게 부여할 수 있는 권한 범위를 줄입니다.
- **수명 주기에 따른 액세스 관리 (SEC03-BP06):** 사용자가 역할을 변경하거나 조직을 떠날 때 그룹 멤버십을 조정하고 액세스를 제거하는 등 조직 내 주체의 수명 주기 전반에 걸쳐 권한을 모니터링하고 조정합니다.
- **공개 및 교차 계정 액세스 분석 (SEC03-BP07):** 공개 및 교차 계정 액세스를 강조하는 결과를 지속적으로 모니터링하고, 이 액세스가 필요한 특정 리소스로만 제한합니다.

탐지

탐지는 예상치 못한 구성 변경과 예상치 못한 동작의 두 부분으로 구성됩니다. 이는 보안 수명 주기의 필수적인 부분이며 잠재적인 보안 잘못된 구성, 위협 또는 예상치 못한 동작을 식별할 수 있게 해줍니다.

- **서비스 및 애플리케이션 로깅 구성 (SEC04-BP01):** 감사, 조사 및 운영 사용 사례를 위해 서비스 및 애플리케이션에서 보안 이벤트 로그를 보관합니다. AWS CloudTrail, VPC Flow Logs, Route 53 확인자 쿼리 로그가 기본 소스입니다.
- **표준화된 위치에 로그, 결과 및 메트릭 캡처 (SEC04-BP02):** 분석을 간소화하기 위해 보안 로그와 결과를 중앙 집중식 로그 아카이브 계정과 같은 표준화된 위치에 캡처합니다. Amazon Security Lake는 데이터를 표준화하고 중앙 집중화하는 데 도움이 될 수 있습니다.
- **보안 경고 상관 관계 분석 및 보강 (SEC04-BP03):** 여러 소스의 보안 경고를 자동으로 상관 관계를 분석하고 추가 정보로 보강하여 사고 식별 및 대응의 정확도를 높입니다. Amazon Detective는 이 프로세스를 지원합니다.
- **비준수 리소스에 대한 수정 시작 (SEC04-BP04):** 구성 요구 사항을 준수하지 않는 리소스에 대해 프로그래밍 방식으로 정의된 수정을 수동 또는 자동으로 시작합니다. AWS Config 및 AWS Security Hub는 비준수 리소스를 탐지하고 수정을 시작하는 데 도움이 될 수 있습니다.

인프라 보호

인프라 보호는 심층 방어와 같은 제어 방법론을 포함하며, 워크로드 내의 시스템과 서비스가 의도하지 않은 무단 액세스 및 잠재적 취약성으로부터 보호되도록 합니다.

네트워크 보호 모범 사례

- **네트워크 계층 생성 (SEC05-BP01):** 데이터 민감도 및 액세스 요구 사항에 따라 워크로드 구성 요소의 논리적 그룹화에 기반하여 네트워크 토폴로지를 다른 계층으로 분할합니다.
- **네트워크 계층 내 트래픽 흐름 제어 (SEC05-BP02):** 필요한 흐름으로만 트래픽을 제한하기 위해 계층 내에서 추가 분할을 사용합니다. 환경으로 들어오고 나가는 트래픽(남-북 트래픽)과 내부 구성 요소 간의 트래픽(동-서 트래픽)을 모두 제어합니다.
- **검사 기반 보호 구현 (SEC05-BP03):** 네트워크 계층 간에 트래픽 검사 지점을 설정하여 전송 중인 데이터가 예상 카테고리 및 패턴과 일치하는지 확인합니다. AWS Network Firewall 및 AWS WAF와 같은 도구를 사용합니다.
- **네트워크 보호 자동화 (SEC05-BP04):** 인프라를 코드로(IaC) 및 CI/CD 파이프라인과 같은 DevOps 방식을 사용하여 네트워크 보호 배포를 자동화하여 변경 사항을 추적하고 배포 시간을 단축합니다.

컴퓨팅 보호 모범 사례

- **취약성 관리 수행 (SEC06-BP01):** 코드, 종속성 및 인프라의 취약점을 자주 스캔하고 패치하여 새로운 위협으로부터 보호합니다. Amazon Inspector 및 AWS Systems Manager Patch Manager를 사용합니다.
- **강화된 이미지에서 컴퓨팅 프로비저닝 (SEC06-BP02):** 강화된 이미지를 배포하여 런타임 환경에 대한 의도하지 않은 액세스 기회를 줄입니다. 신뢰할 수 있는 레지스트리에서만 런타임 종속성을 획득하고 서명을 확인합니다.
- **수동 관리 및 대화형 액세스 축소 (SEC06-BP03):** 가능한 경우 자동화를 사용하여 배포, 구성, 유지 관리 및 조사 작업을 수행합니다. SSH 또는 RDP와 같은 대화형 액세스를 비상 절차로 제한합니다.
- **소프트웨어 무결성 검증 (SEC06-BP04):** 암호화 검증을 사용하여 워크로드에서 사용하는 소프트웨어 아티팩트(이미지 포함)의 무결성을 검증합니다. AWS Signer를 사용하여 코드 서명 수명 주기를 관리합니다.
- **컴퓨팅 보호 자동화 (SEC06-BP05):** 자동화된 스캔을 사용하여 컴퓨팅 리소스 내의 잠재적 문제를 탐지하고 자동화된 프로그래밍 방식 응답 또는 플릿 관리 작업을 통해 수정합니

다.

데이터 보호

데이터 보호는 데이터의 민감도에 따라 분류하고, 저장 데이터와 전송 중인 데이터를 암호화하여 무단 액세스로부터 보호하는 것을 포함합니다.

데이터 분류 및 보호 모범 사례

모범 사례 ID	설명
SEC07-BP01	**데이터 분류 체계 이해:** 워크로드가 처리하는 데이터의 분류, 처리 요구 사항, 관련 비즈니스 프로세스, 저장 위치 및 데이터 소유자를 이해합니다.
SEC07-BP02	**데이터 민감도에 따른 데이터 보호 제어 적용:** 분류 정책에 정의된 각 데이터 클래스에 적절한 수준의 제어를 제공하는 데이터 보호 제어를 적용합니다.
SEC07-BP03	**식별 및 분류 자동화:** 데이터 식별 및 분류를 자동화하여 올바른 제어를 구현하고 인적 오류 위험을 줄입니다. Amazon Macie는 S3 버킷에서 민감한 데이터를 식별하는 데 도움이 될 수 있습니다.
SEC08-BP01	**안전한 키 관리 구현:** 워크로드의 저장 데이터를 보호하는 데 필요한 키 자료의 저장, 교체, 액세스 제어 및 모니터링을 포함하는 안전한 키 관리를 구현합니다. AWS KMS는 이 기능을 제공합니다.
SEC08-BP02	**저장 데이터 암호화 시행:** 저장된 개인 데이터를 암호화하여 기밀성을 유지하고 의도하지 않은 데이터 노출로부터 추가적인 보호 계층을 제공합니다.
SEC08-BP04	**액세스 제어 시행:** 격리 및 버전 관리와 같은 메커니즘을 사용하여 저장 데이터 보호를 위한 액세스 제어를 시행합니다.
SEC09-BP01	**안전한 키 및 인증서 관리 구현:** AWS Certificate Manager(ACM) 및 AWS Private CA와 같은 서비스를 사용하여 네트워크 통신을 보호하고 웹사이트 및 리소스의 ID를 설정하는 TLS 인증서를 관리합니다.
SEC09-BP02	**전송 중 암호화 시행:** 조직의 정책, 규제 의무 및 표준에 따라 정의된 암호화 요구 사항을 시행합니다. 민감한 데이터를 전송할 때는 암호화된 프로토콜만 사용합니다.
SEC09-BP03	**네트워크 통신 인증:** TLS 또는 IPsec과 같이 인증을 지원하는 프로토콜을 사용하여 통신의 ID를 확인합니다.

사고 대응

성숙한 예방 및 탐지 제어 기능을 갖추더라도, 조직은 보안 사고의 잠재적 영향을 완화하고 대응하기 위한 메커니즘을 구현해야 합니다. 준비는 사고 중에 팀이 효과적으로 운영되고, 문제를 격

리, 봉쇄, 포렌식 수행 및 운영을 알려진 양호한 상태로 복원하는 능력에 큰 영향을 미칩니다.

사고 대응 모범 사례

- **핵심 인력 및 외부 리소스 식별 (SEC10-BP01):** 조직이 사고에 대응하는 데 도움이 될 내부 및 외부 인력, 리소스 및 법적 의무를 식별하고 연락처 목록을 유지 관리합니다.
- **사고 관리 계획 개발 (SEC10-BP02):** 사고 대응 프로그램 및 전략의 기초가 되는 공식적인 사고 대응 계획을 개발합니다. 이 계획에는 역할 및 책임, 통신 계획, 사고 대응 단계가 포함되어야 합니다.
- **포렌식 기능 준비 (SEC10-BP03):** 보안 이벤트 조사를 지원하기 위해 포렌식 기능을 개발합니다. 여기에는 관련 로그 및 스냅샷 수집, 데이터 분석 및 결과 보고가 포함됩니다.
- **보안 사고 대응 플레이북 개발 및 테스트 (SEC10-BP04):** 서비스 거부(DoS), 랜섬웨어, 자격 증명 손상과 같은 예상되는 사고 시나리오에 대한 플레이북을 만듭니다.
- **사전 프로비저닝 액세스 (SEC10-BP05):** 사고 대응자가 조사에서 복구까지 걸리는 시간을 줄이기 위해 AWS에 올바른 액세스 권한을 사전에 프로비저닝했는지 확인합니다. 긴급 "break glass" 액세스를 위해 전용 사용자 및 역할을 사용합니다.
- **사전 배포 도구 (SEC10-BP06):** 보안 담당자가 조사에서 복구까지 걸리는 시간을 줄이기 위해 올바른 도구를 사전에 배포했는지 확인합니다. 보안 대응 및 운영 기능을 자동화합니다.
- **시뮬레이션 실행 (SEC10-BP07):** 실제 보안 이벤트 시나리오를 모방한 시뮬레이션(게임 데이)을 실행하여 사고 대응 역량을 평가하고 개선합니다.
- **사고로부터 배우는 프레임워크 구축 (SEC10-BP08):** 각 사고로부터 배우고根本 원인 분석을 수행하여 사고가 재발하는 것을 방지하고 보안 태세를 개선합니다.

애플리케이션 보안(AppSec)

애플리케이션 보안은 개발하는 워크로드의 보안 속성을 설계, 구축 및 테스트하는 전체 프로세스를 설명합니다. 소프트웨어 개발 수명 주기(SDLC)의 일부로 정기적인 애플리케이션 보안 테스트를 채택하면 프로덕션 환경에 보안 문제가 유입되는 것을 방지하는 데 도움이 됩니다.

애플리케이션 보안 모범 사례

- **애플리케이션 보안 교육 (SEC11-BP01):** 팀에 안전한 개발 및 운영 관행에 대한 교육을 제공하여 개발 수명 주기 초기에 보안 문제를 예방, 탐지 및 해결하는 데 도움을 줍니다.

- **개발 및 릴리스 수명 주기 전반에 걸쳐 테스트 자동화 (SEC11-BP02):** 정적 애플리케이션 보안 테스트(SAST) 및 동적 분석 보안 테스트(DAST)와 같은 자동화된 테스트를 사용하여 릴리스 전에 잠재적인 문제를 일관되고 반복적으로 식별합니다.
- **정기적인 침투 테스트 수행 (SEC11-BP03):** 자동화된 테스트나 수동 코드 검토로 탐지할 수 없는 잠재적인 소프트웨어 문제를 식별하기 위해 정기적인 침투 테스트를 수행합니다.
- **코드 검토 수행 (SEC11-BP04):** 개발 중인 소프트웨어의 품질과 보안을 확인하기 위해 코드 검토를 구현합니다. 자동화된 도구를 사용하여 이 프로세스를 지원합니다.
- **패키지 및 종속성 서비스 중앙 집중화 (SEC11-BP05):** 팀이 소프트웨어 패키지 및 기타 종속성을 획득할 수 있는 중앙 집중식 서비스를 제공하여 사용 전에 패키지를 검증할 수 있습니다.
- **소프트웨어를 프로그래밍 방식으로 배포 (SEC11-BP06):** 가능한 경우 소프트웨어 배포를 프로그래밍 방식으로 수행하여 인적 오류로 인한 배포 실패 또는 예상치 못한 문제 발생 가능성을 줄입니다.
- **파이프라인의 보안 속성 정기적 평가 (SEC11-BP07):** 소프트웨어를 빌드하고 배포하는 데 사용하는 파이프라인에 Well-Architected 보안 원칙을 적용하고, 특히 권한 분리에 주의를 기울입니다.
- **워크로드 팀에 보안 소유권을 부여하는 프로그램 구축 (SEC11-BP08):** 빌더 팀이 자신이 만드는 소프트웨어에 대한 보안 결정을 내릴 수 있도록 권한을 부여하는 프로그램이나 메커니즘을 구축합니다.